



#12
3-16-04
C.J.H.
1043

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Applicants:	David J. Matz, et al	§	Art Unit:	2173
		§		
Serial No.:	09/342,765	§		
		§	Examiner:	Omar A. Kazmi
Filed:	June 29, 1999	§		
		§		
Title:	Portable User Interface	§	Docket No.	ITL.0215US
	For Presentation Of	§		(P7121)
	Information Associated	§		
	With Audio/Video Data	§		

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

RECEIVED
MAR 08 2004
Technology Center 2100

APPEAL BRIEF

Dear Sir:

Applicant hereby appeals from the Final Rejection dated September 23, 2003.

I. REAL PARTY IN INTEREST

The real party in interest is Intel Corporation, the assignee of the present application by virtue of the assignment recorded at Reel/Frame 010069/0644.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

03/05/2004 CNGUYEN 00000075 09342765

01 FC:1402

330.00 OP

Date of Deposit February 26, 2004
I hereby certify under 37 CFR 1.8(a) that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage on the date indicated above and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.
Janice Munoz
Janice Munoz

III. STATUS OF THE CLAIMS

The application was originally filed with claims 1-20. Claims 21-31 were added during the prosecution of the application. Claims 1-31 have been finally rejected and are the subject of this appeal.

IV. STATUS OF AMENDMENTS

There are no unentered amendments.

V. SUMMARY OF THE INVENTION

Referring to Fig. 1, an embodiment of a system 8 is shown which may be a computer or any other type of device suitable for presenting audio and/or video data, such as set-top boxes, game systems, and the like. The system 8 may include a browser 10 that is capable of accessing information on an electronic communications network, such as the Internet or other type of network, including local area networks (LANs) and wide area networks (WANs). The browser 10 may be any conventional browser such as Netscape Navigator from Netscape Communications Corporation, Internet Explorer from Microsoft Corporation, or other browsers. As used here, the term "browser" includes any program, routine, module, or component (implemented in software or hardware or a combination of the two) that is capable of accessing locations on a communications network (e.g., the Internet, LAN, WAN, and so forth). In the ensuing description, although reference is made to communicating over the Internet, it is contemplated that the browser 10 may communicate over other networks in further embodiments.

Specification, pp. 2-3.

The browser 10 is able to locate pages of information on the Internet based on a uniform resource locator (URL). To view a desired page, a user may enter a URL in the browser 10 or choose a link to a URL from a currently displayed page. The browser 10 may then download the desired page. A page may include a text file written in a markup language such as Standard Generalized Markup Language (SGML), an SGML subset including Extensible Markup Language (XML), or an SGML application such as Hypertext Markup Language (HTML). Other types of markup languages are also available. A version of HTML is described in "HTML 4.0 Specification," W3C Recommendation, dated April 1998. An overview of SGML guidelines is presented in an article entitled "A Gentle Introduction to SGML" at the web site {<http://www-tei.uic.edu/orgs/tei/sgml/index.html>}. A version of XML is described in "Extensible Markup Language (XML™) 1.0," World Wide Web Consortium (W3C) Recommendation, dated February 1998, and available at {<http://www.w3.org>}. Specification, p. 3.

An HTML file typically includes markup instructions regarding how text is to be displayed in a browser window 11. The HTML file may prescribe several characteristics of the text including the font, point size, color, and location within the browser window 11. Conventional HTML files may also provide information regarding the display of graphics and images. The browser 10 may thus display text and graphics in the browser window 11 according to instructions in the HTML file. Specification, p. 3.

The browser 10 may download an HTML file 12 over path 2 from a source (e.g., a location on the Internet, a location on another type of network, or a local storage device

in the system 8). In addition to instructions specifying display of text and images, the HTML file 12 may also be accompanied by a script 13, which is a program or collection of instructions that may be part of the HTML file 12 or separate from but associated with the HTML file 12. An HTML file that is associated with a script may be referred to as a scripted HTML file. The script 13 may be executed in response to a number of events, including: loading of the HTML file 12, a user action in the browser 10, receipt of message from another module in the system, and so forth. The scripting language used to create the scripts is flexible and can be user-specified. Although reference is made to a singular script 13 in the described embodiment, it is contemplated that multiple scripts may be attached to the HTML file 12 in further embodiments. Specification, pp. 3-4.

In accordance with some embodiments, the script 13 may be executable at the time the HTML file 12 is loaded to create user interface components 14 for an interface in the browser window 11. The user interface components 14 include display components 14A to present displayable information associated with multimedia data stored in a source 18. The user interface components also include control components 14B to control presentation of the multimedia data in the source 18. As used here, the term “multimedia data” may include audio and/or video data (hereinafter “audio/video data”). Audio/video data may include audio and video data as well as one of audio and video data. Displayable information associated with the audio/video data may include statistical information about the audio/video data (e.g., time remaining, track number, language, etc.), status information, and other related information. Further, the displayable information may also include video images. The multimedia data stored in

the source 18 may be stored on a CD, a DVD, or some other storage source (e.g., a hard disk drive or other storage element in the system 8 or a storage device located over a network). Specification, p. 4.

Referring to Fig. 4, an example arrangement of the user interface components 14 in the browser window 11 may include a statistics display area 40 that has a time remaining indicator to show the amount of time remaining in a music track or film, a track indicator to indicate the current track, and other statistical information. Further, status indicators may provide indications of the status of the source 18 (e.g., whether it is playing, stopped, and so forth). In addition, a play area 46 (such as for presenting video images) may be included. Although shown as taking a small portion of the browser window 11, the video play area 46 can be configured to take up most of the browser window 11 or even the entire screen of a monitor or display. As illustrated, the statistics display area 40 and play area 46 are part of the display component 14A. The user interface control components 14B may include control buttons, including a play button 41, a stop button 42, a pause button 43, a fast forward (or forward skip) button 44, and a rewind (or reverse skip) button 45. Activation of any one of the control buttons in the user interface control components 14B generates an event that triggers execution of an event handler in the script 13. A display event may also be generated in the system 8 in response to data (audio, video, and/or related information) transferred from the source 18. Additionally, a display event may be generated in response to the source 18 changing states, e.g., turning on or off, stopping, playing, etc. Specification, pp. 4-5.

The script 13 may include event handlers 15 for responding to the various events that may occur in the system 8, including display events and user input events. Such event handlers in a script when associated with an HTML file may also be referred to as HTML scripted handlers. Control event handlers 15 triggered by activation of corresponding user interface control components 14B are executed to generate control messages for controlling the source 18. One or more display event handlers 15 may be triggered by a display event (generated in response to data transferred from the source 18) to present displayable information relating to the multimedia data contained in the source 18. Specification, p. 5.

A control module 16 provides the interface to the source 18. The control module 16 may include a software routine such as a Java applet or an ActiveX component. A description of Java may be found in Ivor Horton, "Beginning Java," Wrox Press Ltd. (1997). ActiveX components are elements of Microsoft's Component Object Model (COM) that provides a framework for designing reusable software components. A description of the COM model may be found in "The Component Object Model: A Technical Overview," dated October 1994, which may be found at the web site {<http://www.microsoft.com/com/tech/com.asp>}. ActiveX components are described in "Introduction to ActiveX Controls," which may be found at the web site {<http://msdn.microsoft.com/workshop/components/default.asp>}. Specification, p. 5.

An ActiveX component encapsulates the data and subroutines employed to perform specific operations. ActiveX components may be linked to other software applications to perform these operations. Because ActiveX components are designed

with standard interfaces for accessing the operations that the components provide, a single ActiveX component may interact with several different software applications or routines, including the event handlers 15 of the script 13. Thus, in one example, a browser 10 such as Internet Explorer that supports ActiveX control can be used to download a scripted HTML file to create the desired user interface components in the browser window 11 in which information associated with multimedia data in the source 18 may be viewed. The script 13 associated with the loaded HTML file is able to interact with the control module 16 to control presentation of multimedia data in the source 18. By employing the control module 16, the tasks of creating and interacting with the user interface components 14 in the browser window 11 (performed by event handlers 15 in the script 13) may be separated from the tasks of interfacing the source 18 (performed by the control module 16). Consequently, the script 13 including the event handlers 15 may be designed to work with any number of platforms including different types of sources 18. Specification, p. 6.

The control module 16 may be loaded into the system 8 in any of a number of ways. For example, the control module 16 may be stored on magnetic or optical media for transfer into the system 8. Alternatively, the control module 16 may be downloaded by the browser 10 from a location on an electronic communications network. The control module 16 may interact with a device driver (not shown) for the source 18.

Specification, p. 6.

The script associated with a scripted HTML file, or portions of the file, may be easily modifiable by a user using a number of different techniques. Using HTTP over a

network, the entire HTML file or a piece of the HTML file (such as a script associated with the file) may be replaced. HTTP stands for Hypertext Transfer Protocol, which is described in RFC 2068, entitled "Hypertext Transfer Protocol--HTTP/1.1," dated January 1997. This allows the user to customize the user interface components 14 in the browser window 11. Specification, p. 6.

Referring to Fig. 2, activation of one of the control buttons in the user interface control components 14B generates a corresponding user input event (at 30). For example, if the source 18 includes a CD or DVD player that contains a CD or DVD, a user may activate one of the control buttons to play, stop, or perform other control operations. The same or similar control components may be used in further embodiments to control other types of media. In response to the user input event, a corresponding control event handler 15 in the script 13 is triggered to generate control messages that are directed to the control module 16 (at 32). Such control messages may include, for example, messages indicating that the source 18 is to be played, stopped, paused, and so forth. In turn, the control module 16 may transmit corresponding commands (at 34) to the source 18 (through the device driver for the source 18). Thus, for example, a play command from the control module 16 may cause the source 18 to transfer data to the control module 16, including audio/video data, information identifying the playback position measured in reference to the beginning or end of a track, and other information. A stop or pause command may cause the source 18 to stop transferring data to the control module 16. Specification, pp. 6-7.

Referring to Fig. 3, in response to the control module 16 receiving data from the source 18 (at 20), the control module 16 may generate a display event (at 22) that triggers a display event handler 15 in the script 13. Data from the source 18 may accompany the display event, which are processed by the display event handler 15 to determine statistical information, including time remaining, time elapsed, and track number based on the data identifying the current location of a CD or DVD track. A display event may also be generated in response to the source changing states, e.g., turning on or off, starting, stopping, pausing, and so forth. Such a display event may be associated with some indication of the status of the source 18 from which status information of the source 18 may be generated by the display event handler 15. The statistical and status information may be sent to corresponding user interface display components 14 in the browser window 11 (at 24). Specification, p. 7.

Referring to Fig. 5, the system 8 may be a personal computer, a set-top box, or any other type of system capable of presenting multimedia data. In accordance with one embodiment, the system 8 may include several software layers 50 that are executable on a central processing unit (CPU) 51, which may be a microprocessor, microcontroller, or other control device. The software layers may include an operating system, device drivers, and application programs such as the browser 10 and the control module 16 that execute within an operating system environment. The CPU 51 may be coupled to a primary bus 53 through a host bridge 52 to communicate with other components in the system 8. A video controller 54 is also coupled to the primary bus 53 to display text and

video images on a monitor 55. A storage controller 58 controls access to the source 18. Specification, pp. 7-8.

In addition, a network interface 56 is coupled to the primary bus 53 to control communications with a network 57 such as a LAN, a WAN, or the Internet. The system 8 may also include a transceiver 59 coupled to the primary bus 53 through which the system may communicate with another communications link such as a telephone line, satellite link, a cable link, and so forth. Specification, p. 8.

The CPU 51 may interact with other hardware devices connected to a secondary bus 61 through a system bridge 60. Input devices such as a keyboard 63 and a pointer device 64 may allow a user to provide input to the system 8 through an input/output (I/O) controller 62 that is coupled to the secondary bus 61. Additionally, a sound card 67 may be coupled to the secondary bus 61 to present audio signals to a speaker 69. Specification, p. 8.

In one example embodiment, the source 18 in the system 8 may be an audio source that supplies audio data. For example, the audio source may include an optical medium such as a CD or DVD that contains musical tracks or other forms of audio. Alternatively, the audio source may be an audio file located on the Internet or at another storage location, either in the system 8 itself (e.g., a hard disk drive) or in an external storage device accessible by the system 8 (such as over a network). In the embodiment including an audio source, audio data received by the control module 16 from the audio source 18 may be routed to the sound card 67 through the appropriate device driver to present audio signals onto the speaker 69. In addition, statistical and status information

associated with the audio data, e.g., the name or number of the current musical track, whether the musical track is playing or stopped, the time remaining in the musical track, and so forth, as generated by a display event handler 15 may be presented in the statistics area 40 in the browser window 11 (Fig. 4). Thus, in accordance with an embodiment of the invention, while audio is being played in real time by the system 8, a user can view statistical and status information associated with the audio data in the browser window 11. As audio data is being streamed, the display event handler 15 in the script 13 can determine the associated statistical and status information and display them in the browser window 11. Additionally, the user is also able to control the playing or presentation of audio through control buttons in the browser window 11. Specification, pp. 8-9.

In another embodiment, the source 18 may be a video source supplying video data from a video file, which may be located on an optical medium such as a CD or DVD, or some other media. In this embodiment, the user interface display components 14A may include a play area 46 (Fig. 4) in the browser window 11 to display video images from the source 18. Control buttons 41-45 may be used to control presentation of the video images. Further, other control buttons may be available to a user to adjust the display of the video images (e.g., aspect ratio, etc.). Specification, p. 9.

Referring to Fig. 6, in accordance with some embodiments, to view or access multimedia data contained in the source 18, a user can launch (at 100) the browser 10. Next, a scripted HTML file (or some other markup language file with one or more associated scripts) can be loaded (at 102) into the system 8 from an internal or external

storage device. This may trigger (at 104) a setup event handler 15 in the script 13, which sets up user interface components 14 in the browser window 11 including both control and display components. In response to activation (at 106) of one of the user interface control components, a user input control event may be generated. This triggers (at 108) control event handlers 15 in the script 13 to create and send corresponding control messages to the control module 16. If the control message is to begin playing a music or video track in the source 18, the source 18 may respond by transferring data to the control module 16, which in turn generates (at 110) display events to trigger (at 112) the display event handler 15 in the script 13. The display event handler 15 processes the data received from the source 18 as discussed above and presents (at 114) statistical information, status information, and/or video images for presentation in the user interface display components 14A. Specification, p. 9.

Some embodiments of the invention may include one or more of the following advantages. By employing a browser to display information associated with multimedia data such as audio and video data, and to control the source containing the multimedia data, a familiar user interface is provided to users. Convenience may be enhanced since multiple programs do not need to be separately invoked to play a CD or DVD, for example. Modifications of the interfaces in browsers may be more convenient as compared to conventional CD or DVD programs, since the script associated with a scripted HTML file may be more easily modified than source code. As a result, a user may be able to more conveniently customize a user interface in the browser to present information associated with the multimedia data. Further, a scripted HTML file may be

executable on different platforms. Another benefit of some embodiments is that a control module to interface the source containing the multimedia data may be a component-based software module. By utilizing a software module such as an ActiveX control component to control access to a multimedia data source, the operations performed by an event handler such as an HTML scripted handler can be separated from the underlying operations that are performed by the control module. Further, an HTML scripted handler may be used with several different control modules. Also, the control module may be used with several different HTML scripted handlers. Specification, pp. 9-10.

Various software (formed of modules, routines, or other layers, for example), including the several layers discussed above, may be stored or otherwise tangibly embodied in one or more machine-readable storage media in the system 8. Storage media suitable for tangibly embodying software instructions may include different forms of memory including semiconductor memory devices such as dynamic or static random access memories, erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs), and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; and optical media such as CD or DVD disks. The instructions stored in the one or more storage media when executed cause the system 8 to perform programmed acts. Specification, p. 10.

The software can be loaded into the system 8 in one of many different ways. For example, instructions or other code segments stored on one or more storage media or transported through a network interface card, modem, or other interface mechanism may

be loaded into the system 8 and executed to perform programmed acts. In the loading or transport process, data signals that are embodied as carrier waves (transmitted over telephone lines, network lines, wireless links, cables and the like) may communicate the instructions or code segments to the system 8. Specification, p. 10.

VI. ISSUES

- A. Can claims 1-9 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 1?
- B. Can claims 10 and 11 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 10?
- C. Can claims 12-15 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 12?
- D. Can claims 16-19 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 16?
- E. Can claim 20 be anticipated when the cited reference fails to teach all claim limitations?
- F. Can claims 21-29 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 21?
- G. Can claims 30 and 31 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 30?

VII. GROUPING OF THE CLAIMS

Claims 1-9 can be grouped together; claims 10 and 11 can be grouped together; claims 12-15 can be grouped together; claims 16-19 can be grouped together; claims 21-29 can be grouped together; claims 30 and 31 can be grouped together; and claim 20 is separately patentable for the reasons set forth below. With this grouping, all claims of a particular group stand or fall together. Furthermore, regardless of the grouping that is set forth by the Examiner's rejections, the claims of each group set forth in this section stand alone with respect to the claims of the other groups that are set forth in this section. In other words, any claim of a particular group that is set forth in this section does not stand or fall together with any claim of any other group that is set forth in this section.

VIII. ARGUMENT

All claims should be allowed over the cited references for the reasons set forth below.

A. **Can claims 1-9 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 1?**

The system of independent claim 1 includes a source, a browser and a markup language file. The source contains multimedia data, and the browser has a user interface to display a presentation of the multimedia data. The markup language file is associated with a script handler and is loadable by the browser. The script handler is executable to

process multimedia data that is received from the source for presentation to the browser user interface.

The Examiner rejects independent claim 1 under 35 U.S.C. § 103(a) in view of the combination of U.S. Patent No. 6,188,401 (herein called "Peyer") and U.S. Patent No. 5,572,643 (herein called "Judson"). Peyer generally describes a script-based user interface system. Figure 4 of Peyer depicts an example of the system disclosed in Peyer. More specifically, Figure 4 depicts an Internet content window 152, a live television window 154 and a user interface component 150. Peyer states, "all three components are implemented as browser instances." Peyer, 6:37-43. Furthermore, Peyer discloses that television or video content is preferably viewed in an ActiveX control. Peyer, 6:32-33. Peyer further adds, "such a control uses its own browser instance, and runs as a window on top of any other content." Peyer, 6:34-35. Thus, Peyer generally describes a system that has a first browser instance for a user interface and a separate, second browser instance for a television content window. Thus, Peyer teaches separate browser instances to display multimedia data and a user interface.

Judson generally discloses a link process to a server in which information objects are output through a viewer. Judson states that the information objects cover "all forms of messages, notices, texts, graphics, sound, video, tables, diagrams, applets and other content, and any combinations of any of the above." Judson, 7:38-43. Furthermore, Judson states that the information object may include "an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof." Judson, 8:8-11. However, Judson fails to teach or suggest a script

handler that is executable to process multimedia data that is presented to a browser user interface.

Thus, neither reference cited by the Examiner teaches or suggests the markup language file of independent claim 1. More specifically, neither reference teaches or suggests a markup language file that is associated with a script handler and is loadable by a browser, where the script handler is executable to process multimedia data received from a source for presentation to a browser user interface. To summarize, Peyer does not teach or suggest that its user interface displays multimedia data. Rather, Peyer teaches away from such an arrangement, in that Peyer teaches separate browser instances for each of these functions. *See, for example*, Peyer, 6:6-35. As acknowledged by the Examiner, Judson fails to teach a script handler that is executable to process multimedia data. Final Office Action, 4.

Therefore, for at least the reason that the combination of references fails to teach or suggest all claim limitations, a *prima facie* case of obviousness has not been established for independent claim 1.

A *prima facie* case of obviousness has not been established for independent claim 1 for at least the independent reason that the Examiner fails to show where the prior art contains the alleged suggestion or motivation to combine Peyer and Judson to derive the claimed invention. In fact, Peyer teaches away from a script handler to process multimedia data that is presented to a browser user interface and thus, teaches away from the claimed invention, thereby negating any suggestion or motivation to modify Peyer to derive the claimed invention. M.P.E.P. § 2145.X.D. Furthermore, Peyer teaches the

advantage of maintaining ActiveX controls in a separate interface for controlling the television content. Therefore, the proposed modification would improperly destroy the principal of operation of Peyer's system, thereby providing an additional independent reason why a *prima facie* case of obviousness has not been established for independent claim 1. M.P.E.P. § 2145.X.D.

The Examiner fails to show where the prior art contains the alleged suggestion or motivation for the combination of Peyer and Judson to derive the claimed invention. More specifically, for a *prima facie* case of obviousness, the Examiner must show that one skilled in the art, *without knowledge of the claimed invention*, would have combined Peyer and Judson to derive the claimed invention. The Examiner cites *In re Fine*, 5 USPQ2d 1596 (Fed. Cir. 1988), a Federal Circuit case that holds that a suggestion or motivation must exist in the prior art in order for there to be a *prima facie* case of obviousness. However, contrary to the holding in this case, the Examiner fails to show where the prior art contains this alleged suggestion or motivation. It is noted that in *In re Fine*, the Examiner's rejection was overturned for not showing where the prior art contained the alleged suggestion or motivation for the modification of a reference. Thus, the Examiner must specifically point out in a reference where the prior art contains the alleged suggestion or motivation for the combination of references to derive the claimed invention. See *Ex parte Gambogi*, 62 USPQ2d 1209, 1212 (Bd. Pat. App. & Int. 2001); *In re Rijckaert*, 28 USPQ2d 1955, 1957 (Fed. Cir. 1993); M.P.E.P. § 2143.

Therefore, a *prima facie* case of obviousness has not been established for independent claim 1.

Claims 2-9 are patentable for at least the reason that these claims depend from an allowable claim. Thus, the § 103(a) rejections of claims 1-9 are improper and should be reversed.

B. Can claims 10 and 11 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 10?

The system of independent claim 10 includes a source that contains audio/video data, a browser and a file that is associated with predetermined instructions. The browser has an user interface to display a presentation of the audio/video data; and the file is loadable by the browser. The instructions are executable to display information that is associated with the audio/video data in the source.

The Examiner rejects independent claim 10 under 35 U.S.C. § 103(a) in view of the combination of Peyer and Judson. However, neither reference teaches nor suggests a browser that has a user interface to display a presentation of audio/video data.

More specifically, Peyer clearly teaches separate instances of a browser program for purposes of displaying television content and user controls. Furthermore, Judson fails to teach or suggest the missing claim limitations. Therefore, for at least the reason that the combination of references fails to teach or suggest all claim limitations, a *prima facie* case of obviousness has not been established for independent claim 10.

Additionally, Peyer teaches away from the claimed invention, in that Peyer teaches separate browser instances for the user controls and television content. Modifying Peyer, as set forth by the Examiner's rejection, would destroy the principal of operation of Peyer's invention. These factors are evidence that there is no suggestion or

motivation in art for the combination of Peyer and Judson. Furthermore, the Examiner fails to make a showing to the contrary, as the Examiner fails to show where the prior art contains the alleged suggestion or motivation for the combination of these two references to derive the claimed invention.

Therefore, a *prima facie* case of obviousness has not been established for independent claim 10.

Claim 11 is patentable for at least the reason that this claim depends from an allowable claim. Therefore, the § 103(a) rejections of claims 10 and 11 are improper and should be reversed.

C. Can claims 12-15 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 12?

The method of independent claim 12 is for displaying information that is associated with multimedia data. The method includes loading a markup language file that is associated with a script handler and invoking the script handler to create a user interface in a browser. The method includes displaying the information that is associated with the multimedia data with the user interface in the browser and displaying a presentation of the multimedia data in the browser.

The Examiner rejects independent claim 12 under 35 U.S.C. § 103(a) in view of the combination of Peyer and Judson. However, neither reference fails to teach or suggest displaying a presentation of multimedia data in a browser and displaying information associated with multimedia data with a user interface in a browser. Instead,

Peyer teaches a user interface in one browser instance and a television window in another browser instance. Thus, information that is associated with multimedia data is not displayed in a user interface in a browser in Peyer. Furthermore, Judson fails to teach or suggest the missing claim limitations. Thus, a *prima facie* case of obviousness has not been established for claim 12 for at least the reason that the combination of Peyer and Judson fails to teach or suggest all claim limitations.

Additionally, a *prima facie* case of obviousness has not been established for independent claim 12 for the additional, independent reason that the Examiner fails to show where the prior art contains the alleged suggestion or motivation for the combination of Peyer and Judson to derive the claimed invention. The disclosure of Peyer, to the contrary, contradicts such a suggestion or motivation in prior art. Peyer teaches away from the claimed invention, and the modification that is proposed by the Examiner's rejections would destroy the principal of operation of Peyer's invention.

Thus, a *prima facie* case of obviousness has not been established for independent claim 12.

Claims 13-15 are patentable for at least the reason that these claims depend from an allowable claim. Therefore, the § 103(a) rejections of claims 12-15 are improper and should be reversed.

D. Can claims 16-19 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 16?

The method of independent claim 16 includes loading a file into a browser and creating an interface in the browser based on instructions that are associated with the file. The method includes receiving multimedia data from a source and displaying information that is associated with the multimedia data in the browser interface and presentation of the multimedia data in the browser interface.

The Examiner rejects independent claim 16 under 35 U.S.C. § 103(a) in view of the combination of Peyer and Judson. However, neither one of these references teaches nor suggests displaying a presentation of multimedia data in a browser interface, where an interface is created in the browser based on instructions associated with a file. Judson does not address displaying a presentation of multimedia data, as acknowledged by the Examiner. Final Office Action, 4. Furthermore, Peyer teaches separate instances for displaying multimedia presentations and a user interface and thus, does not teach or suggest the missing claim limitations. Therefore, for at least the reason that the combination of references fails to teach or suggest all claim limitations, a *prima facie* case of obviousness has not been established for independent claim 16.

A *prima facie* case of obviousness has not been established for independent claim 16 for at least the additional, independent reason that the Examiner fails to show where the prior art contains the alleged suggestion or motivation for the combination of Peyer and Judson. More specifically, the Examiner must show that one skilled in the art, *without knowledge of the claimed invention*, would have combined Peyer and Judson to

derive the claimed invention. Because the Examiner has failed to make this showing, a *prima facie* case of obviousness has not been established for independent claim 16 for at least this additional, independent reasons.

Claims 17-19 are patentable for at least the reason that these claims depend from an allowable claim. Thus, the § 103(a) rejections of claims 16-19 are improper and should be reversed.

E. Can claim 20 be anticipated when the cited reference fails to teach all claim limitations?

The article of independent claim 20 includes one or more machine-readable storage media that store instructions for presenting audio/video data. The instructions when executed cause a system to generate an interface in a browser. The interface is created by a scripted language file. The instructions when executed cause the system to receive multimedia data from a source, display information that is associated with the multimedia data and the interface of the browser and present the multimedia data in the browser.

The Examiner rejects independent claim 20 under 35 U.S.C. § 102(e) in view of Peyer. However, the Examiner fails to show where Peyer allegedly teaches instructions to cause a system to generate an interface in a browser (where the interface is created by a scripted markup language file) and present multimedia data in the browser. Instead, Peyer clearly teaches a browser instance to display multimedia presentation and *another* browser instance that provides a user interface. *See, for example*, Figure 4 of Peyer that

shows the user interface 152 and a live television window 154, each of which is implemented as a separate browser instance. Peyer, 6:37-39.

Thus, because Peyer fails to teach all limitations of independent claim 20, Peyer fails to anticipate this claim. Therefore, the § 102(e) rejection of independent claim 20 is improper and should be reversed.

F. Can claims 21-29 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 21?

The system of independent claim 21 includes a browser that has a user interface and a markup language file that is associated with a script handler. The markup language file is loadable by the browser, and the script handler is executable to process multimedia data that is received from a source for presentation to the browser user interface.

The Examiner rejects independent claim 21 under 35 U.S.C. § 103(a) in view of the combination of Peyer and Judson. However, Peyer teaches a user interface that is displayed in a separate browser instance than a multimedia presentation. Therefore, Peyer fails to teach or suggest the limitations of claim 21 relied on by the Examiner. Furthermore, modifying Peyer in view of Judson so that the user interface of Peyer displays a multimedia presentation would improperly change the principal of operations of Peyer's invention, as Peyer clearly teaches the advantage of using ActiveX controls in a separate user interface. Additionally, Peyer teaches away from such a modification for similar reasons. Furthermore, the Examiner fails to specifically show where the prior art

contains the alleged suggestion or motivation to combine Peyer and Judson to derive the claimed invention.

Therefore, for at least any of these reasons, a *prima facie* case of obviousness has not been established for independent claim 21.

Claims 22-29 are patentable for at least the reason that these claims depend from an allowable claim. Thus, the § 103(a) rejections of claims 21-29 are improper and should be reversed.

G. Can claims 30 and 31 be rendered obvious when the Examiner has failed to establish a *prima facie* case of obviousness for independent claim 30?

The system of independent claim 30 includes a browser that has a user interface and a file that is associated with predetermined instructions. The file is loadable by the browser, and the instructions are executable to display information that is associated with audio/video data in a source.

The Examiner rejects independent claim 30 under 35 U.S.C. § 103(a) in view of the combination of Peyer and Judson. However, contrary to the limitations of independent claim 30, Peyer teaches away from a file that is loadable by a browser (that has a user interface) so that instruction to the file may be executed to display information that is associated with audio/video data in a source. To the contrary, Peyer teaches separate browser instances to display a user interface and audio/video data. Thus, the modification that is proposed by the Examiner's rejection is improper, as this rejection changes the principal of operation of Peyer's invention and is further improper for at least

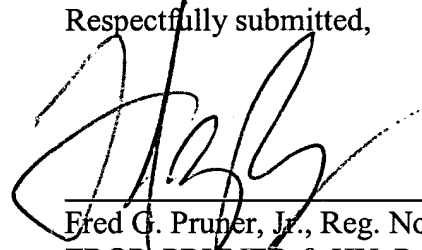
the additional, independent reason that Peyer teaches away from such a modification. Additionally, the Examiner fails to specifically show where the prior art contains the alleged suggestion or motivation to combine Peyer and Judson to derive the claimed invention. Therefore, for at least any of these reasons, a *prima facie* case of obviousness has not been established for independent claim 30.

Claim 31 is patentable for at least the reason that this claim depends from an allowable claim. Thus, the § 103(a) rejections of claims 30 and 31 are improper and should be reversed.

IX. CONCLUSION

Applicant requests that each of the final rejections be reversed and that the claims subject to this appeal be allowed to issue.

Respectfully submitted,



Date: February 26, 2004

Fred G. Pruner, Jr., Reg. No. 40,779
TROP, PRUNER & HU, P.C.
8554 Katy Freeway, Suite 100
Houston, TX 77024-1805
713/468-8880 [Phone]
713/468-8883 [Facsimile]

APPENDIX OF CLAIMS

The claims on appeal are:

1. A system comprising:

a source containing multimedia data;

a browser having a user interface to display a presentation of the multimedia data;

and

a markup language file associated with a script handler and loadable by the browser, the script handler executable to process multimedia data received from the source for presentation to the browser user interface.
2. The system of claim 1, wherein the markup language file includes a Hypertext Markup Language file.
3. The system of claim 1, wherein the source includes a compact disc drive.
4. The system of claim 1, wherein the source includes a digital video disc drive.
5. The system of claim 1, further comprising a control module adapted to provide an interface to the source.
6. The system of claim 5, wherein the control module includes an ActiveX component.

7. The system of claim 1, wherein the browser is capable of interfacing with an ActiveX component.

8. The system of claim 1, wherein the user interface includes one or more user interface control components, and wherein the script handler is triggered in response to activation of a user interface control component.

9. The system of claim 1, wherein the user interface includes one or more user interface display components to display information associated with the multimedia data.

10. A system comprising:
a source containing audio/video data;
a browser having a user interface to display a presentation of the audio/video data;
and
a file associated with predetermined instructions, the file loadable by the browser and the instructions executable to display information associated with the audio/video data in the source.

11. The system of claim 10, wherein the displayed information includes a status of the source.

12. A method of displaying information associated with multimedia data, comprising:
loading a markup language file associated with a script handler;
invoking the script handler to create a user interface in a browser;
displaying the information associated with the multimedia data with the user
interface in the browser; and
displaying a presentation of the multimedia data in the browser.
13. The method of claim 12, further comprising accessing the multimedia data stored
in a storage source through a control module.
14. The method of claim 13, wherein the control module includes an ActiveX
component.
15. The method of claim 12, wherein the user interface includes one or more control
components, the method further comprising receiving activation of a user interface control
component to control a source containing the multimedia data.
16. A method of displaying multimedia data, comprising:
loading a file into a browser;
creating an interface in the browser based on instructions associated with the file;
receiving multimedia data from a source;
displaying information associated with the multimedia data in the browser
interface; and

displaying a presentation of the multimedia data in the browser interface.

17. The method of claim 16, wherein the file includes a Hypertext Markup Language file.

18. The method of claim 16, further comprising accessing the multimedia using a control module.

19. The method of claim 18, wherein the control module includes an ActiveX component.

20. An article including one or more machine-readable storage media storing instructions for presenting audio/video data, the instructions when executed causing a system to:

generate an interface in a browser, wherein the interface is created by a scripted markup language file;

receive multimedia data from a source;

display information associated with the multimedia data in the interface of the browser; and

present the multimedia data in the browser.

21. A system comprising:
a browser having a user interface; and
a markup language file associated with a script handler and loadable by the browser, the script handler executable to process multimedia data received from a source for presentation to the browser user interface.
22. The system of claim 21, wherein the markup language file includes a Hypertext Markup Language file.
23. The system of claim 21, wherein the source includes a compact disc drive.
24. The system of claim 21, wherein the source includes a digital video disc drive.
25. The system of claim 21, further comprising a control module adapted to provide an interface to the source.
26. The system of claim 25, wherein the control module includes an ActiveX component.
27. The system of claim 21, wherein the browser is capable of interfacing with an ActiveX component.

28. The system of claim 21, wherein the user interface includes one or more user interface control components, and wherein the script handler is triggered in response to activation of a user interface control component.

29. The system of claim 21, wherein the user interface includes one or more user interface display components to display information associated with the multimedia data.

30. A system comprising:
a browser having a user interface to display a presentation of the audio/video data;
and
a file associated with predetermined instructions, the file loadable by the browser and the instructions executable to display information associated with audio/video data in a source.

31. The system of claim 30, wherein the displayed information includes a status of the source.